

# Análisis de Eficiencia de Algoritmos. Simulación como Recurso de Enseñanza-Aprendizaje

---

Edith Lovos<sup>1</sup>, Tatiana Gibelli<sup>1</sup>

<sup>1</sup> UNRN, Av. Don Bosco y Leloir s/n, R8500FAL, Viedma, Argentina

La resolución de un problema usando una computadora puede ser modelada por diferentes algoritmos. De aquí la importancia de contar con herramientas que nos permitan seleccionar el algoritmo que sea más eficiente, es decir, el que insuma menos recursos del ordenador: tiempo y memoria necesarios para su ejecución. Existen dos formas de medir la complejidad de un algoritmo: análisis teórico y análisis empírico.

En este trabajo se presenta la descripción del simulador, "SIMULA-ALGO", junto con la propuesta de uso y detalles del diseño de la interfaz. Se presenta además una fundamentación teórica sobre las ventajas del uso de la simulación como recurso pedagógico-didáctico.

El simulador se propone como recurso para la enseñanza aprendizaje del análisis teórico de eficiencia de algoritmos. El mismo está destinado a alumnos de los primeros años de carreras universitarias de licenciatura en sistemas. Los contenidos considerados en el simulador, requieren, para su comprensión, del uso de procesos cognitivos analíticos. El simulador pretende optimizar estos procesos en el alumno que lo usa. Así, por ejemplo, el simulador podrá realizar procesos rutinarios en forma rápida y efectiva, permitiendo que el alumno se concentre en los procesos analíticos que requiere la tarea.

*Palabras clave:* Simulación, Algoritmos, Enseñanza, Universidad

## 1. Introducción

### 1.1 Algoritmos y análisis de complejidad

Cuando se quiere resolver un problema usando una computadora es necesario, en primer lugar, realizar un modelado (abstracción) de la situación. Uno de los pasos fundamentales es organizar los elementos y datos del problema. Para ello se utilizan diferentes estructuras de datos. De esta manera es posible organizar el conjunto de datos elementales del problema con el objetivo de facilitar su manipulación.

En segundo lugar, es necesario determinar el algoritmo, es decir la secuencia de pasos que conducen a la resolución del problema y luego poder implementar estos pasos en un lenguaje de programación. Un algoritmo se caracteriza por: estar definido por pasos claros y concretos, tener un número finito de pasos, ser preciso (cada paso da un único resultado) y ser independiente del lenguaje de programación y de la máquina donde se lo ejecute.

Finalmente, un problema puede ser resuelto por diferentes algoritmos. Así, es muy importante seleccionar el algoritmo que sea más eficiente, el que insuma menos recursos del ordenador: tiempo y memoria necesarios para su ejecución. Por tanto, es necesario poder medir el consumo de recursos de un programa para establecer su nivel de calidad [1].

Existen dos formas de medir la complejidad de un algoritmo: análisis teórico y análisis empírico. El simulador propuesto en este trabajo, se basa en el análisis teórico.

El análisis teórico, establece una medida intrínseca de la cantidad de trabajo realizado por el algoritmo que permite comparar y seleccionar la mejor implementación. Nos interesa estudiar el comportamiento de los algoritmos para grandes volúmenes de datos, por ello se usa la notación asintótica (para hacer referencia a la velocidad de crecimiento de los valores de una función). Así, la evaluación de la eficiencia involucra además el uso de representaciones gráficas de funciones, cuya lectura requiere capacitación por parte de los alumnos.

### 1.2 La simulación

Según Ausubel [2] el aprendizaje significativo tiene lugar cuando una nueva información "se conecta" con un concepto relevante pre-existente en la estructura cognitiva, esto implica que las nuevas ideas, conceptos y proposiciones pueden ser aprendidos significativamente en la medida en que otras ideas, conceptos o proposiciones relevantes estén

adecuadamente claras y disponibles en la estructura cognitiva del individuo y que funcionen como un punto de "anclaje" a las primeras.

El uso de la simulación por computadora, es una técnica de enseñanza que facilita el aprendizaje significativo. Una simulación es una representación de un sistema real. Cualquier simulación debe reunir las siguientes características:

- Imitar la realidad, pero

No ser real en sí misma

Permitir que los usuarios efectúen cambios sobre la misma

Así, los simuladores son programas que contienen modelos del mundo real. La acción básica del aprendiz es realizar cambios en las variables de entrada y observar las consecuencias sobre las variables de salida. Hennessy[3]

La resolución de problemas de cualquier tipo requieren, para su comprensión, del uso de procesos cognitivos analíticos. "La enseñanza y el aprendizaje basados en la resolución de problemas dispone de un valioso auxiliar mediante la práctica simulada de situaciones reales con valores propedéutico; facilita el procesamiento de la información; brinda ocasiones para la práctica, la adquisición o mejora del ritmo de aprendizaje, la sistematización y la transferencia de lo aprendido. ". Malbrán, Pérez [4].

El proceso de ejecución de un algoritmo, realizado por la computadora, es un proceso complejo, desarrollado por la máquina en forma rápida y eficiente. Es importante que el alumno, "futuro programador", pueda comprender como se comportan los algoritmos, según la estructura de datos seleccionada (arreglos, listas, árboles) para organizar la información (datos) y los métodos de resolución empleados (recursividad por ejemplo). La comprensión de este proceso requiere del uso de abstracción. Uno de los objetivos del simulador propuesto es facilitar los procesos cognitivos del alumno para la comprensión de estos temas.

## 2. Propuesta de uso

### 2.1 Contenidos y Objetivos

El simulador propuesto, que denominaremos "SIMULA-ALGO", constituye un recurso para la enseñanza – aprendizaje que incluye los temas: Estructuras de datos, algoritmos y análisis de eficiencia.

El objetivo de la propuesta de uso del simulador es brindar a los alumnos un recurso que les permita:

- Visualizar (mediante modelos gráficos) las diferentes estructuras de datos con las que se trabaja en la materias Programación I y II

Observar el comportamiento de diferentes algoritmos, según la estructura de datos seleccionada (arreglos, listas, árboles) para organizar la información (datos) y los métodos de resolución empleados (recursividad por ejemplo).

Visualizar la complejidad de los algoritmos y cómo varía ésta según el tamaño de las entradas del mismo.

Adquirir un entrenamiento que les permita identificar, ante un problema dado, cuál es la estructura que mejor se adapta al problema y cual es el algoritmo más eficiente para resolver los requerimientos del mismo, además de inferir conclusiones para la aplicación en el diseño de otras estructuras complejas.

Lograr una comprensión significativa del comportamiento asintótico de funciones mediante la aplicación de este tema a una situación concreta vinculada a la carrera: el cálculo de la eficiencia del algoritmo.

En el Mapa Conceptual de la figura 1, se resume la información sobre contenido, opciones y uso del simulador.

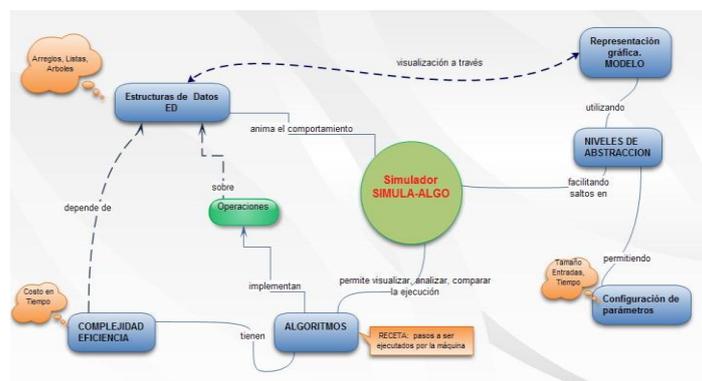


Figura 1. Mapa Conceptual

### 2.2 Ámbito de Aplicación

El ámbito de aplicación del simulador es el nivel universitario. Se propone su uso en las materias Programación I y II del primer año de la Lic. en Sistemas de la Universidad Nacional de Río Negro (UNRN) [5].

El simulador estará disponible en el aula virtual de apoyo a las materias de programación mencionadas. El entorno utilizado será la plataforma Moodle [6].

El material podrá ser utilizado tanto en clases prácticas de las materias de programación, como así también el alumno podrá continuar su entrenamiento en tiempo extra-áulico.

## 2.3 Destinatarios

El simulador SIMULA-ALGO, esta dirigido a los alumnos universitarios de primer año, los cuales han egresado recientemente del nivel medio, cuyas edades oscilan entre 18 y 25 años. Es importante destacar que estos usuarios, tienen manejo de interfaces como la propuesta en este trabajo.

Los destinatarios de este simulador deberán contar con conocimientos de resolución de problemas, algoritmos, y en el cálculo de complejidad de algoritmos.

Como así también, con conocimientos de la asignatura Matemática tales como funciones, sus representaciones gráficas y su comportamiento asintótico (funciones lineales, cuadráticas, logarítmicas).

## 3. Fundamentación del recurso

### 3.1 Fundamentación cognitiva

Los contenidos considerados en el simulador, requieren, para su comprensión, del uso de procesos cognitivos analíticos que constan de "componentes" y "metacomponentes" (para buscar información pertinente, procesarla y controlar el propio pensamiento, "metacognición"). Los mismos están presentes en cualquier proceso de resolución de problemas. El simulador pretende optimizar estos procesos en el alumno que lo usa. Así, por ejemplo, el simulador podrá realizar procesos rutinarios (cálculo de la complejidad, gráfico de las funciones que representan el orden de complejidad, etc) en forma rápida y efectiva, permitiendo que el alumno se concentre en los procesos analíticos que requiere la tarea.

Por otra parte el uso de las abstracción es un proceso inherente a los contenidos considerados. La resolución de un problema utilizando la computadora (por medio de algoritmos) requiere un pensamiento analógico pues se debe utilizar una representación de la realidad a través de un modelo. En este sentido, el simulador pretende facilitar la comprensión utilizando un lenguaje gráfico ad-hoc para representar los conceptos abstractos presentes en los algoritmos (tales como punteros, árboles, etc.). El mismo tiene además la ventaja de ser "dinámico", es decir, que le permite ir visualizando los cambios en el modelo gráfico.

El análisis de complejidad de los algoritmos requiere articular información y generar una síntesis personal. Los procesos de síntesis constituyen uno de los tres aspectos de la inteligencia humana, concebida como procesamiento de la información (Sternberg, subteoría Experiencial, "Inteligencia Creativa")[7]. La interpretación de las gráficas de las funciones que muestran la complejidad de los algoritmos que provee el simulador, requieren procesos cognitivos de alto nivel,

entre ellos, procesos de síntesis visual y comparación selectiva.

El simulador proveerá, en distintas instancias, información adicional que constituye un refuerzo a la memoria a corto plazo, permitiendo de esta forma que el usuario avance en el desarrollo de las actividades y concentre su esfuerzo en otras actividades cognitivas implicadas en la tarea.

Los mensajes de error, y otras instancias de feedback previstas en el simulador constituyen una ayuda a los procesos metacognitivos y autorregulatorios, proporcionándole al usuario instancias de control y análisis de los propios procesos cognitivos implicados en la tarea. Los procesos metacognitivos también se facilitan con la posibilidad de configurar los parámetros (como velocidad de ejecución del algoritmo) y solicitar información adicional, herramientas que el usuario podrá utilizar de acuerdo a su grado de experticia

El simulador brindará además la posibilidad de experimentar distintas posibilidades (estructuras de datos, tamaño de las entradas, algoritmos, etc), buscando de esta forma facilitar el entrenamiento y la posibilidad de establecer relaciones y elaborar conclusiones, así como conjeturar que ocurre en situaciones o problemas "análogos".

### 3.2 Justificación de la estructura

El simulador propuesto presenta un recorrido en orden secuencial. Esto se debe a las características intrínsecas del contenido y la metodología de trabajo con el mismo. Las distintas instancias del recorrido del simulador son análogas a los pasos que se siguen para la resolución de un problema usando una computadora. Para ello se necesita modelar un algoritmo (receta, conjunto de pasos que guían la solución), el cual será implementado (traducido) en un lenguaje de programación dado. Este algoritmo, procesa información la cual va a ser almacenada en alguna estructura de datos, la elección de ésta determina la forma de acceso a los elementos que contiene (por ejemplo en una lista el acceso es secuencial). Luego la complejidad o eficiencia se mide sobre un determinado algoritmo, ya que la misma cambia según las instrucciones del algoritmo. Es decir, dos algoritmos pueden arrojar el mismo resultado, pero uno puede ser más o menos eficiente que otro, en función de la optimización del uso de los recursos (tiempo de cpu, memoria), y esta optimización depende de la forma que tenga el algoritmo y de la estructura de datos seleccionada.

## 4. Detalles de la implementación

### 4.1 Diagrama de Navegación

En la página de inicio del simulador se podrá acceder a un tutorial que presenta el modo de uso y posibilidades del recurso. Esto se realizará a través de un vídeo con ejemplos. Por otra parte, durante el recorrido del simulador se brindan vínculos (ayuda) para acceder a información que refresque los conocimientos y conceptos utilizados.

El contenido del simulador estará organizado en forma jerárquica, y el acceso al mismo será en forma secuencial:

- 1 En primer lugar, el alumno deberá seleccionar la estructura de datos a utilizar y “cargar” la misma (ingresar los datos). El simulador brindará una visualización gráfica del mismo a través de un “modelo”.

En segundo lugar, el alumno deberá elegir el algoritmo que aplicará sobre la estructura seleccionada anteriormente. El simulador mostrará los pasos de dicho algoritmo permitiendo regular al usuario el tiempo de ejecución, dependiendo del grado de experticia que tenga en el tema. La elección del desarrollo “paso a paso” del algoritmo le permite ir observando el recorrido por los distintos “pasos” y visualizando cómo se aplican esas operaciones sobre la estructura (causa-efecto). Este modo es recomendable para usuarios novatos.

Finalmente, el usuario podrá consultar la eficiencia del algoritmo aplicado. Así, el simulador mostrará el orden de eficiencia y el tiempo de ejecución del mismo de acuerdo al tamaño de las entradas del algoritmo. Se permitirá además realizar una comparación de la eficiencia del algoritmo respecto a otros, tanto en forma analítica, mediante una tabla que muestra el incremento del tiempo de ejecución de acuerdo al orden; como también en forma gráfica, mediante la visualización de las gráficas correspondientes a los diferentes órdenes. En ambos casos, se resaltarará con un color diferente, el orden de ejecución del algoritmo analizado para facilitar la comparación.

Este proceso podrá repetirse y simularse con diferentes estructuras de datos sobre el mismo algoritmo y viceversa, de manera que pueda observar y analizar los cambios en la eficiencia del algoritmo.

Puede verse en el siguiente gráfico (Figura 2) un “diagrama de navegación” que muestra el acceso a las distintas vistas y opciones del simulador.

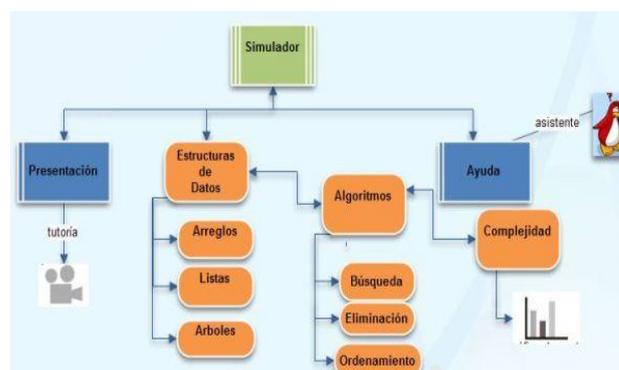


Figura 2: Mapa de Navegación

### 4.2 Interfaz gráfica del simulador

La interfaz del usuario del simulador SIMULA-ALGO estará compuesta por un menú de opciones y un conjunto de ventanas (Figura 3) a las cuales se podrá acceder, a medida que el usuario avanza en la simulación. A éstas ventanas, las denominaremos vistas, entre las que podemos distinguir:

- una vista para la representación gráfica de la estructura de datos a evaluar (VED),
- una vista que muestra las instrucciones del algoritmo (escrito en el lenguaje de programación Pascal) (VA),

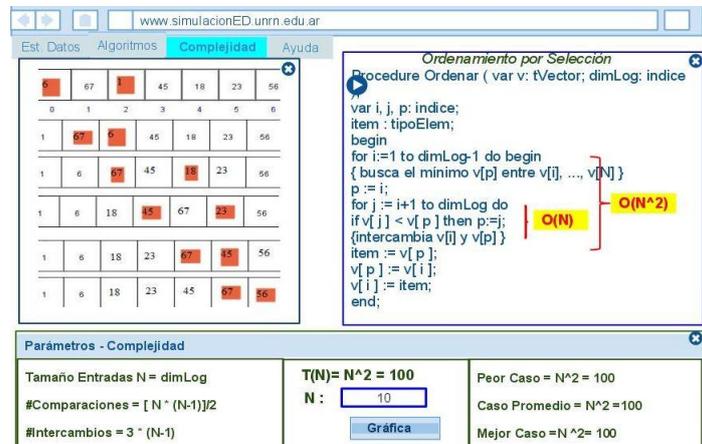


Figura 3: Interface gráfica

- una vista para los parámetros que entran en juego en la simulación del algoritmo seleccionado (VP).
- una vista con la representación gráfica del comportamiento del algoritmo (función asintótica) (VFA).

En las vistas mencionadas se utilizarán distintos atributos para los objetos (estructuras de datos). Uno de los atributos más utilizados en este simulador será el color.

La vista VED, mostrará el comportamiento del algoritmo, a medida que se ejecuten cada una de las

instrucciones que lo componen en la VA. De manera dinámica en esta vista se mostrará que instrucción se ejecuta y cuál es la representación gráfica ('efecto') asociada a dicha ejecución en la vista de VED.

Durante la ejecución del algoritmo, en la VED se podrán apreciar cambios en la estructura de datos seleccionada, los mismos están diferenciados a través del uso de colores. Por ejemplo, en la ejecución del Algoritmo de Ordenamiento por Selección, que mostrará con colores los elementos al intercambiar en cada pasada (Ver figura 3).

### 4.3 Iconografía

A continuación se hace una breve descripción de los distintos íconos utilizados:

Menú de selección: permite el acceso a las distintas ventanas del simulador.

Personaje asistente del simulador: Se utiliza la imagen del Pingüino "H" que acompaña el recorrido del simulador y provee ayuda en las distintas instancias, brindando la posibilidad de acceder a información que complementa y "refresca" los conocimientos necesarios para la comprensión del proceso simulado. El mismo se encuentra presente en las distintas vistas del simulador.

Ventanas de diálogo: Las mismas proveen información adicional y mensajes de error.

Botones interactivos: los mismos usan íconos intuitivos que permiten realizar acciones, algunas de las cuales están asociadas a atajos de teclado. Entre los principales botones interactivos utilizados podemos destacar:

- botón interactivo de cierre de ventana: permite cerrar las ventanas que el usuario no desee visualizar.

botón interactivo de inicio de la ejecución del algoritmo seleccionado e indicador del avance: permite ejecutar el algoritmo, y en la visualización "paso a paso", indica el estado de avance en el recorrido del algoritmo (el "paso" que se está simulando actualmente).

## Conclusiones

El simulador propone una alternativa de enseñanza-aprendizaje frente a métodos tradicionales. Estos últimos, en general, implican el uso de recursos estáticos como el pizarrón, una presentación digital, y de animaciones a través de algún software visualizador de estructuras de datos que presentan gráficos para representar los conceptos abstractos con los que trabajan los algoritmos (arreglos, listas, árboles, etc). Para este último caso, existen algunos productos de software, como es el caso de Visualization [8], que muestra una animación sobre el comportamiento de algunas de las estructuras de datos, y de algunos algoritmos de

operaciones básicas sobre esas estructuras de datos, pero no permite visualizar las instrucciones de los algoritmos ni la complejidad de los mismos. Frente a esta alternativa, el simulador propuesto presenta las siguientes ventajas:

- La posibilidad de mostrar adecuadamente la ejecución de un algoritmo, que constituye un concepto de carácter dinámico. Con este recurso el proceso es simulado "paso a paso" mostrando el comportamiento del algoritmo sobre la estructura de datos elegida, y el código (conjunto de instrucciones - pasos) que lo describe y en cada paso de la simulación se va mostrando (visualizando) el resultado de la ejecución de los pasos del algoritmo a través de la representación gráfica del efecto asociado a la misma.

Atender a un grupo heterogéneo de alumnos respecto al ritmo de comprensión, pues el simulador permite al alumno visualizar este proceso a una velocidad que puede ser configurada por el usuario de manera que puede analizar cada paso de la ejecución del algoritmo y visualizar los efectos del mismo.

Permite que el aprendiz pueda "probar" distintas posibilidades, tanto de estructuras (con diferente cantidad de datos) como algoritmos, pudiendo analizar su complejidad. Así se da lugar a la posibilidad de error del usuario (elegir un algoritmo poco eficiente o una estructura inadecuada) sin que sufra las consecuencias que tendría desarrollar o aplicar un algoritmo ineficiente en determinadas condiciones reales.

El usuario tiene la posibilidad además de analizar la complejidad del algoritmo, mediante la observación del gráfico que representa la eficiencia y comparar además con otros posibles algoritmos.

- Los cálculos y gráficos que ofrece el simulador (que son realizados en forma eficiente y rápida) permiten al alumno concentrarse en procesos cognitivos de alto nivel, entre ellos, procesos analíticos, de síntesis visual y de comparación selectiva.

A través del uso del simulador propuesto se busca proveer los alumnos y docentes de las asignaturas de programación de los primeros años de una herramienta para la enseñanza-aprendizaje de estructuras de datos, algoritmos y análisis de la eficiencia de los mismos. Todos estos conceptos fundamentales en la carrera Lic. en Sistemas.

## Agradecimientos

Agradecemos los aportes realizados por el Prof. Gustavo Marincoff durante el cursado de la materia Psicología Cognitiva de la maestría Tecnología

Informática Aplicada en Educación de la Facultad de  
Informática de la Universidad Nacional de la Plata.

*Dirección de Contacto del Autor/es:*

## Referencias

- [1] Abad Soriano María Teresa, Análisis de la Eficiencia de los Algoritmos. Departamento de Lenguajes y Sistemas Informáticos. FIB-UPC. Universidad Politécnica de Catalunya. (2010). <http://www.lsi.upc.edu/~ada/apunts/MTA/efi.pdf>. Consultado en octubre 2010.
- [2] Ausubel -Novak -Hanesian, Psicología Educativa: Un punto de vista cognoscitivo .2º Ed. TRILLAS México (1983)
- [3] Hennessy, S.: "Learner perceptions of realism and magic in computer simulations". En: British Journal of Educational Technology, (2003).
- [4] Malbran M.C y Pérez, V. Potencialidad Educativa de la simulación mediada por ordenadores. X Congreso Argentino de Computación (CACIC 2004)
- [5] UNRN – Universidad Nacional de Río Negro [www.unrn.edu.ar](http://www.unrn.edu.ar).
- [6] Moodle – <http://moodle.org/>
- [7] Sternberg, R. J., Beyond IQ: A Triarchic Theory of Intelligence. Cambridge: Cambridge University Press, (1985)
- [8] VISUALIZATION <http://www.cs.usfca.edu/~galles/visualization/>

**Edith Lovos**  
Av. Don Bosco y Leloir s/n  
R8500FAL  
Viedma  
Argentina  
e-mail: [elovos@unrn.edu.ar](mailto:elovos@unrn.edu.ar)  
sitio web: <http://www.unrn.edu.ar>

**Tatiana, Gibelli**  
Av. Don Bosco y Leloir s/n  
R8500FAL  
Viedma  
Argentina  
e-mail: [tgibelli@unrn.edu.ar](mailto:tgibelli@unrn.edu.ar)  
sitio web: <http://www.unrn.edu.ar>

---

**Ing. Edith Lovos** es Prof. Adjunta en la UNRN en la Lic. De Sistemas de la Sede Atlántica y becaria de postgrado del programa de 'Becas de Formación Conducentes a la Formación de Postgrado' de la UNRN.

---

---

**Lic. Tatiana Gibelli** es Profesora Adjunta en la UNRN en el área matemáticas y estadísticas de la Sede Atlántica. Estudiante de postgrado de la maestría TIAE de la Facultad de Informática de la UNLP.

---